

# Designing Games to Handle Coupled Constraints

Na Li and Jason R. Marden

**Abstract**—The central goal in multiagent systems is to design local control laws for the individual agents to ensure that the emergent global behavior is desirable with respect to a given system level objective. In many systems, such as cooperative robotics or distributed power control, the design of these local control algorithms is further complicated by additional coupled constraints on the agents' actions. There are several approaches in the existing literature for designing such algorithms stemming from classical optimization theory; however, many of these approaches are not suitable for implementation in multiagent systems. This paper seeks to address the design of such algorithms using the field of game theory. Among other things, this design choice requires defining a local utility function for each decision maker in the system. This paper seeks to address the degree to which utility design can be effective for dealing with these coupled constraints. In particular, is it possible to design local agent utility functions such that all pure Nash equilibrium of the unconstrained game (i) optimize the given system level objective and (ii) satisfy the given coupled constraint. This design would greatly simplify the distributed control algorithms by eliminating the need to explicitly consider the constraints. Unfortunately, we illustrate that designing utility functions within the standard game theoretic framework is not suitable for this design objective. However, we demonstrate that by adding an additional state variable in the game environment, i.e., moving towards state based games, we can satisfy these performance criteria by utility design. We focus on the problem of consensus control to illustrate these results.

## I. INTRODUCTION

Many engineering systems can be characterized as a large scale collection of interacting subsystems where each one makes local independent decisions in response to local information. The central challenge in these multiagent systems is to derive desirable collective behaviors through the design of individual agent control algorithms [1]–[7]. The potential benefits of distributed decision architectures include the opportunity for real-time adaptation and robustness to dynamic uncertainties.

In many systems the desirable collective behavior must satisfy a given coupled constraint on the agents' actions [2]–[7]. One example is the problem of TCP control where the users' sending rates need to satisfy link capacity constraints [4], [8]. Alternatively, in the problem of power control in MIMO interference systems, the sending rates need to satisfy service quality constraints [5], [6]. Regardless of the

specific application domain, these coupled constraints bring additional complexity to the control algorithm design.

There are two main research directions aimed at designing distributed control algorithms to satisfy performance criteria that involve coupled constraints. The first direction seeks to design algorithms that ensure the coupled constraint is always satisfied. The well studied consensus algorithm in [3], [9] is an example.<sup>1</sup> While theoretically appealing, algorithms like the consensus algorithm require synchronous updates/clocks amongst the agents and lack a robustness to environmental uncertainties and noisy measurements. The second direction seeks to design algorithms to ensure that *only* the limiting behavior satisfies the coupled constraints, e.g., [4], [7], [10]. While there are efficient algorithms to achieve this, such as dual decomposition, these algorithms often require a two-time scale solution approach by introducing additional pricing terms. Depending on application domains, this two-time scale approach may be prohibitive.

Recently, game theory has emerged as a viable design paradigm for distributed control in multiagent systems. A game theoretic approach to cooperative control involves (i) modeling the agents as self-interested decision makers in a game theoretic environment and (ii) specifying a distributed learning algorithm, i.e., a distributed control law, that ensures that emergent global behavior is desirable with respect to a given system level objective [2], [5], [7], [11], [12]. Our primary focus in this paper is modeling the agents in a game theoretic where we focus specifically on the design of local agent utility functions. There are many objectives that factor into the design of agent utility functions ranging from locality of information to the existence and efficiency of the resulting stable solutions, i.e., (pure) Nash equilibria [12]–[14]. In this paper we focus on whether utility design can be an effective tool for dealing with these coupled constraints in multiagent systems.

Before highlighting the details of our approach, it is important to note that games with coupled constraints has been extensively studied in the game theoretic literature for the past 50 years. However, the research direction is fundamentally different than the one presented in this paper. The key difference is highlighted by the role of utility functions. In the existing literature both the agent utility function and the coupled constraints are *inherited*. Therefore, research focuses

This research was supported by AFOSR grant #FA9550-09-1-0538 and the Social and Information Sciences Laboratory at Caltech.

Na Li is a graduate student of Control and Dynamical Systems, California Institute of Technology, Pasadena, CA 91125, USA [nali@caltech.edu](mailto:nali@caltech.edu)

J. R. Marden is with the Department of Electrical, Computer & Energy Engineering, University of Colorado at Boulder, Boulder, CO 80309, USA [jason.marden@colorado.edu](mailto:jason.marden@colorado.edu)

<sup>1</sup>In the consensus problem, initiated in [1], a group of agents seek to achieve an agreement, or consensus, upon a common value by repeatedly interacting with one another. The consensus algorithm ensures that all agents reach consensus on the (weighted) average of every agent's initial value by forcing the (weighted) average of the agents' values to be invariant along the trajectory.

on establishing conditions that ensure the existence of a Nash equilibrium [15]–[17]. To a lesser extent there has been a degree of work centered on identifying distributed procedures for finding such equilibria [7], [18], [19]. However most of these procedures only work in restricted settings.<sup>2</sup> Among other things, this research highlights the inherent complexity associated in dealing with games with coupled constraints.

The key difference between the research direction proposed in this paper and the classical research on coupled constraint games stems from the fact that in engineering systems utility functions are *designed* as opposed to inherited. Our focus in this paper is whether utility design be an effective tool for dealing with coupled constraints in multiagent systems. For concreteness, consider any multiagent system with a given system level objective and an associated coupled constraint. Is it possible to design agent utility functions for an *unconstrained* game such that *all* resulting equilibria not only satisfy the system level objective but also satisfy the coupled constraint? Among other benefits, such a design would permit the utilization of many established game theoretic tools that are unavailable for coupled constraint games.

The contribution of this work is twofold and primarily focuses on the problem of consensus control: First, we establish an impossibility result demonstrating that designing agent utility functions within the standard game theoretic framework is not suitable for handling systems with coupled constraints. This means that in general it is impossible to design agent utility function such that all resulting Nash equilibria satisfy both the system level objective and the desired coupled constraints. Second, we develop a methodology for constructing games to handle coupled constraints using an extension of the standard game theoretic framework, termed state based games, which introduces an underlying state space into the game structure. We illustrate our methodology on the weighted average consensus problem and demonstrate that this approach ensures that all resulting equilibria of the designed unconstrained game optimize our system level objective and at the same time satisfy our given coupled constraints. While the theoretical developments in this paper hold only for the consensus problem, the approach can easily be extended to alternative classes of problems.

## II. BACKGROUND: NONCOOPERATIVE GAMES

A noncooperative game consists of a set of players  $\mathcal{N} := \{1, 2, \dots, n\}$  where each player  $i \in \mathcal{N}$  has an action set  $\mathcal{A}_i$  and a cost function  $J_i : \mathcal{A} \rightarrow \mathbb{R}$  where  $\mathcal{A} := \mathcal{A}_1 \times \mathcal{A}_2 \times \dots \times \mathcal{A}_n$  is referred to as the set of joint action profiles.<sup>3</sup> For an action profile  $a = (a_1, a_2, \dots, a_n)$ , let  $a_{-i}$  denotes the action profile of players other than player  $i$ , i.e.,  $a_{-i} = (a_1, \dots, a_{i-1}, a_{i+1}, \dots, a_n)$ . An action profile  $a^* \in$

$\mathcal{A}$  is called a *pure Nash equilibrium* if for all players  $i \in \mathcal{N}$ ,

$$J_i(a_i^*, a_{-i}^*) = \min_{a_i \in \mathcal{A}_i} J_i(a_i, a_{-i}^*).$$

In distributed engineering systems a pure Nash equilibrium, or just equilibrium, represents a stable operating point.

In this paper we consider the case when there are coupled constraints on the joint action profiles. This means that the set of *feasible* joint action profiles is constrained to the set  $\mathcal{C} \subseteq \mathcal{A}$ . For any feasible action profile  $a \in \mathcal{C}$ , we denote the set of feasible actions for player  $i$  as  $\mathcal{C}_i(a_{-i}) \subseteq \mathcal{A}_i$ . A feasible action profile  $a^* \in \mathcal{C}$  is called a constrained (pure) Nash equilibrium if for all players  $i \in \mathcal{N}$ ,

$$J_i(a_i^*, a_{-i}^*) = \min_{a_i \in \mathcal{C}_i(a_{-i}^*)} J_i(a_i, a_{-i}^*).$$

## III. A WORKING EXAMPLE: THE CONSENSUS PROBLEM

In this section, we introduce the consensus problem as an example of a multiagent system where the desirable collective behavior must satisfy a coupled constraint on the agents' individual behaviors. We review existing approaches and discuss their respective limitations. Lastly, we demonstrate that designing cost functions within the framework of noncooperative games to ensure that all resulting equilibrium satisfy both our system level objective and our desired coupled constraint is impossible.

### A. Formulation of the Consensus Problem

In the consensus problem initiated in [1] we have a set of agents  $\mathcal{N} = \{1, \dots, n\}$  and each agent  $i$  has an initial value  $v_i(0) \in \mathbb{R}^m$ . This value could represent a measurement, a physical location, or a belief about a particular event. The goal of the consensus problem is to establish an iterative procedure that allows each agent to reach agreement on a common value which in this paper takes on the form of a weighted average of the initial value, i.e., for each agent  $i$ ,

$$\lim_{t \rightarrow \infty} v_i(t) = \frac{\sum_{i \in \mathcal{N}} w_i v_i(0)}{\sum_{i \in \mathcal{N}} w_i} := v^*,$$

where  $v_i(t)$  represents agent  $i$ 's value at stage  $t$  and  $w_i > 0$  is the relative weight of agent  $i$ .

The information available to each agent is represented by an undirected and connected graph  $G = (\mathcal{N}, \mathcal{E})$  with nodes  $\mathcal{N}$  and edges  $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ . We refer to this graph as the interaction graph. By convention we assume that  $(i, i) \in \mathcal{E}$  for each agent  $i$ . Define the neighbors of agent  $i$  as  $\mathcal{N}_i := \{j \in \mathcal{N} : (i, j) \in \mathcal{E}\}$ . We restrict our attention to local algorithms of the following form:

$$v_i(t) = \mathcal{F}_i \left( \{v_j(0), v_j(1), \dots, v_j(t-1), w_j\}_{j \in \mathcal{N}_i} \right),$$

i.e., the value of each agent at any time  $t$  can only depend on the previous values of the agent's neighbors.

<sup>2</sup>This class of algorithms typically involve a two-time scale process where constraints are translated to costs by analyzing the associated dual problem.

<sup>3</sup>We use the terms players and agents interchangeably. Furthermore, we use the term cost functions instead of utility functions as this is the convention for cost minimization systems.

## B. Prior Work

There has been a large body of work analyzing this weighted average consensus problem using both game theoretic and non-game theoretic tools. One of the most prevalent non-game theoretic algorithms, known as the “consensus algorithm” [3], takes on the form

$$v_i(t) = v_i(t-1) + \frac{\epsilon}{w_i} \sum_{j \in \mathcal{N}_i} (v_j(t-1) - v_i(t-1))$$

where  $\epsilon > 0$  is the step-size. When using this algorithm, the weighted average is invariant i.e., for all times  $t$ ,

$$\sum_{i \in \mathcal{N}} w_i v_i(t) = \sum_{i \in \mathcal{N}} w_i v_i(0), \quad (1)$$

hence if agents reach consensus on a common value, this value must represent the weighted average. Notice that keeping this weighted average invariant requires that all agents update synchronously and have perfect information regarding the current values of neighboring agents. In many large scale distributed systems these demands are not practical.

One of the first game theoretic approaches to the consensus problem, introduced in [20], investigates the special case of average consensus ( $w_i = 1/n$ ). This approach involves modeling the agents in a game theoretic environment where each agent is assigned a cost function of the form

$$J_i(v_i, v_{-i}) := \sum_{j \in \mathcal{N}_i} \|v_i - v_j\|_2^2. \quad (2)$$

While this design permits extra flexibility in the learning process by no longer requiring consistent clock rates amongst the agents, it has the following limitations. First of all, any common value  $v'$ , i.e.,  $v_i = v'$  for all  $i \in \mathcal{N}$ , is a Nash equilibrium. Furthermore, if we restrict the joint action (value) space using the coupled constraint in (1), every feasible action profile is a constrained Nash equilibrium because any unilaterally deviation violates the constraint. Therefore, providing convergence to the weighted average is not possible using this design.

## C. Limitations of Noncooperative Game Designs

In this section, we explore whether alternative cost designs to (2) could guarantee that all resulting Nash equilibria achieve the desired performance and the coupled constraint. We focus on all local anonymous cost functions of the form

$$J_i(v_i, v_{-i}) := F\left(\{v_j, v_j(0), w_j\}_{j \in \mathcal{N}_i}\right) \quad (3)$$

meaning that a agent’s cost function could depend on the current values, initial values, and weights of all the neighbors. The function  $F(\cdot)$  defines each agent’s cost and is invariant to specific indices assigned to agents. The following example demonstrates that it is impossible to design cost functions of the form (3) that guarantee all Nash equilibria achieve the desired coupled constraint for general consensus problems.

**Example 1.** Consider two weighted average consensus problems with unit weights for each agent. Figure 1 highlights the interaction graph and the initial values (either A or B)

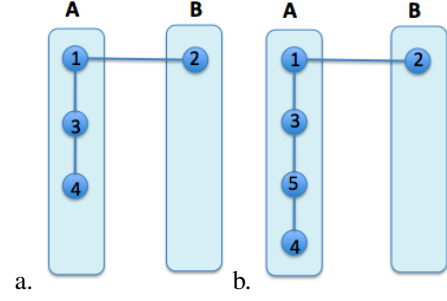


Fig. 1. Interaction Graph for Consensus Problem in Example 1

for two problems. For example, in problem (a) we have the following:  $\mathcal{N} = \{1, 2, 3, 4\}$ ,  $\mathcal{E} = \{\{1, 2\}, \{1, 3\}, \{3, 4\}\}$ ,  $\{v_1(0), v_2(0), v_3(0), v_4(0)\} = \{A, B, A, A\}$ . Define a cost function for each agent of the form (3) and use this same  $F(\cdot)$  for both situations (a) and (b). Because of the anonymous structure of  $F(\cdot)$ , it is straightforward to show that if  $v^{(a)} = (v^*, v^*, v^*, v^*)$  is a Nash equilibrium of (a) then  $v^{(b)} = (v^*, v^*, v^*, v^*)$  is also a Nash equilibrium of (b). The impossibility comes from the fact that  $v^*$  cannot be the weighted average for both situation (a) and (b).

## IV. A STATE-BASED GAME DESIGN FOR CONSENSUS

The previous section demonstrates that the framework of noncooperative game is not suitable for handling coupled constraints through the design of agent cost functions. In this section, we demonstrate that we can overcome the limitations by conditioning cost functions on an additional state variable.

### A. Defining state based games

In this paper we consider a simplified setting of state based games as introduced in [12], [14]. State based games, a simplification of Markov games [21], can be viewed as an extension of noncooperative game by introducing an underlying state space  $X$ . In the state based games considered in this paper, each player  $i$  has a state invariant action set  $\mathcal{A}_i$  and a state dependent cost function  $J_i : X \times \mathcal{A} \rightarrow \mathbb{R}$ . Lastly, there is a deterministic state-transition function of the form  $f : X \times \mathcal{A} \rightarrow X$ .

A state based game proceeds as follows. Let the state at time  $t \in \{0, 1, \dots\}$  be denoted by  $x(t) \in X$ . At any time  $t$ , each player  $i$  myopically selects an action  $a_i(t) \in \mathcal{A}_i$  using only information about the player’s one-stage cost  $J_i(x(t), \cdot)$  and possibly a model for the expected behavior of all other players. For example, if a player used a myopic Cournot adjustment process then  $a_i(t) \in \arg \min_{a_i \in \mathcal{A}_i} J_i(x(t), a_i, a_{-i}(t-1))$ . The state  $x(t)$  and the action profile  $a(t) := (a_1(t), \dots, a_n(t))$  together determine each player’s cost  $J_i(x(t), a(t))$  at time  $t$ . After all players selects their respective action, the ensuing state  $x(t+1)$  is chosen according to the deterministic transition function  $x(t+1) = f(x(t), a(t))$  and the process is repeated.

Before defining our notion of equilibrium for these state based games, we introduce the notion of reachable states. For an state-action pair  $[x^0, a^0]$ , the set of reachable states by an action invariant state trajectory is defined as

$$\bar{X}(x^0, a^0; f) := \{x^0, x^1, x^2, \dots\}$$

where  $x^{k+1} = f(x^k, a^0)$  for all  $k \in \{0, 1, \dots\}$ . Using this notion of reachable states, we define state based Nash equilibria as:

**Definition 1.** (State based Nash equilibrium) The action state pair  $[x^*, a^*]$  is a state based Nash equilibrium if for every player  $i \in \mathcal{N}$  and every state  $x \in \bar{X}(x^*, a^*; f)$ ,

$$J_i(x, a_i^*, a_{-i}^*) = \min_{a_i \in \mathcal{A}_i} J_i(x, a_i, a_{-i}^*) \quad (4)$$

From this definition, we have the following lemma.

**Lemma 1.** If  $[x^*, a^*]$  is a state based Nash equilibrium, then  $\forall \bar{x} \in \bar{X}(x^*, a^*; f)$ ,  $[\bar{x}, a^*]$  is also a state based Nash equilibrium.

### B. Our design

The key to our proposed design is to utilize the additional state as the slack variable to transfer the couple constraint from the action profiles as in (1) to a combination of the action profile and the state. This design permits agents to violate the coupled constraint on the agents' values while the introduced state, termed *bias*, acts in similar fashion to a memory variable that can be utilized to ensure that all equilibrium satisfy the coupled constraint. This bias term takes on a role similar to a Lagrange multiplier.

The starting point of our design is an underlying state space  $X$  where each state  $x \in X$  is defined as a tuple  $x := (v, b)$  where  $v = (v_1, \dots, v_n) \in \mathbb{R}^{mn}$  is the profile of values and  $b = (b_1, \dots, b_n) \in \mathbb{R}^{mn}$  is the profile of bias terms where  $b_i \in \mathbb{R}^m$ . Each agent  $i$  is assigned an action set  $\mathcal{A}_i$  that permits agents to change their value and change their bias by passing their bias to neighboring agents. Specifically, an action is defined as a tuple  $a_i = (\hat{v}_i, \hat{b}_i)$  where  $\hat{v}_i$  indicates a change in value and  $\hat{b}_i = \{\hat{b}_i^j\}_{j \in \mathcal{N}_i}$  indicates a change in bias where  $\hat{b}_i^j$  indicates the bias that agent  $i$  passes to agent  $j$ . We represent the state transition function  $f(x, a)$  by a set of local state transition functions  $\{f_i^v(x, a)\}_{i \in \mathcal{N}}$  and  $\{f_i^b(x, a)\}_{i \in \mathcal{N}}$ . For a state  $x = (v, b)$  and an action  $a = (\hat{v}, \hat{b})$  the state transition function takes on the form

$$\begin{aligned} f_i^v(x, a) &= v_i + \hat{v}_i \\ f_i^b(x, a) &= b_i - \hat{v}_i + \frac{1}{w_i} \sum_{j \in \mathcal{N}_i} (\hat{b}_i^j - \hat{b}_j^i). \end{aligned} \quad (5)$$

It is straightforward to show that for any initial state  $x^0 = (v^0, b^0 = \mathbf{0})$  and any action trajectory  $a^0, a^1, \dots$ , the resulting state trajectory  $x^k = (v^k, b^k) = f(x^{k-1}, a^{k-1})$  for all  $k \geq 1$  satisfies the following equality:

$$\sum_{i=1}^n w_i v_i^k + \sum_{i=1}^n w_i b_i^k = \sum_{i=1}^n w_i v_i^0. \quad (6)$$

The goal of our design is to establish local state based cost functions such that any state based Nash equilibrium  $[x, a]$  where  $x = (v, b)$  satisfies  $v_i = v^*$  and  $b_i = \mathbf{0}$ . Consider the following cost functions

$$J_i(x, a) = J_i^v(x, a) + J_i^b(x, a) \quad (7)$$

where

$$\begin{aligned} J_i^v(x, a) &= \sum_{j \in \mathcal{N}_i} (w_i + w_j) \|v_i + \hat{v}_i - (v_j + \hat{v}_j)\|_2^2 \\ J_i^b(x, a) &= \sum_{j \in \mathcal{N}_i} w_j \left\| b_j - \hat{v}_j + \frac{1}{w_j} \sum_{k \in \mathcal{N}_j \setminus \{j\}} (\hat{b}_k^j - \hat{b}_j^k) \right\|_2^2 \end{aligned}$$

Let  $\tilde{x} = (\tilde{v}, \tilde{b}) = f(x, a)$ . Simplifying, we have

$$\begin{aligned} J_i(x, a) &= J_i(\tilde{x}, \mathbf{0}), \\ &= \sum_{j \in \mathcal{N}_i} (w_i + w_j) \|\tilde{v}_i - \tilde{v}_j\|_2^2 + \sum_{j \in \mathcal{N}_i} w_j \|\tilde{b}_j\|_2^2 \end{aligned} \quad (8)$$

Notice that the cost function in (7) has a component corresponding to value disagreement and another component corresponding to bias disagreement. The following theorem establishes that all state based Nash equilibria are optimal, i.e., each agent has a value that corresponds to the weighted average and a zero bias.

**Theorem 2.** If the weighted average consensus problem is modeled as a state based game as depicted in Section IV-B, then the state-action pair  $[x, a] = [(v, b), (\hat{v}, \hat{b})]$  is a state based Nash equilibrium if and only if for all agents  $i \in \mathcal{N}$ ,  $v_i = v^*$ ,  $b_i = \mathbf{0}$ ,  $\hat{v}_i = \mathbf{0}$ , and  $\sum_{j \in \mathcal{N}_i} (\hat{b}_i^j - \hat{b}_j^i) = \mathbf{0}$ .

*Proof:* Without loss of generality, we prove this theorem in the one-dimensional ( $m = 1$ ) case, i.e.  $v_i \in \mathbb{R}$ . The proof for the multi-dimensional case is virtually identical but requires some additional notations.

( $\Leftarrow$ ) It is straightforward to verify that any state-action pair  $[x, a] = [(v, b), (\hat{v}, \hat{b})]$  where  $\forall i \in \mathcal{N}$ ,  $v_i = v^*$ ,  $b_i = \mathbf{0}$ ,  $\hat{v}_i = \mathbf{0}$ , and  $\sum_{j \in \mathcal{N}_i} (\hat{b}_i^j - \hat{b}_j^i) = 0$  is a state based Nash equilibrium by verifying the conditions of Definition 1.

( $\Rightarrow$ ) Proving this direction is more complicated, but essentially it employs the same idea. If  $[x, a] = [(v, b), (\hat{v}, \hat{b})]$  is a state based Nash equilibrium then no agent has a unilateral incentive to deviate from the action  $a = (\hat{v}, \hat{b})$ , i.e.,

$$J_i((v, b), (\hat{v}, \hat{b})) \leq J_i((v, b), ((\hat{v}'_i, \hat{v}_{-i}), (\hat{b}'_i, \hat{b}_{-i}))) \quad (9)$$

for any  $\hat{v}'_i \neq \hat{v}_i$  and  $\hat{b}'_i \neq \hat{b}_i$ . Rather than focus on the set of all possible deviations, we focus on two particular types of deviations: (i) an agent changes the value but does not pass the bias, or (ii) an agent passes the bias to one neighbor but does not update the value. It turns out that focusing purely on these two scenarios is enough to ensure that all equilibria satisfy the desired performance criterion.

**Case 1:** Consider the situation where  $\hat{v}'_i = \hat{v}_i + (\hat{v}_i)'$  and  $\hat{b}'_i = \hat{b}_i$ . Defining  $a'_i = (\hat{v}'_i, \hat{b}'_i)$ , we have

$$J_i(x, a'_i, a_{-i}) - J_i(x, a) = \alpha_i \left[ (\hat{v}_i)' - \frac{\beta_i}{\alpha_i} \right]^2 - \frac{\beta_i^2}{\alpha_i}$$

where  $\alpha_i := w_i + \sum_{j \in \mathcal{N}_i} (w_i + w_j) > 0$ , and  $\beta_i := w_i \hat{b}_i - \sum_{j \in \mathcal{N}_i} (w_i + w_j) (\tilde{v}_i - \tilde{v}_j)$ . If  $[x, a]$  is a state based Nash equilibrium, then according to (9) we have

$$\alpha_i \left[ (\hat{v}_i)' - \frac{\beta_i}{\alpha_i} \right]^2 - \frac{\beta_i^2}{\alpha_i} \geq 0 \quad (10)$$

**Case 2:** Consider the situation where  $\hat{v}_i' = \hat{v}_i$  and  $\hat{b}_i' = (\hat{b}_i^j + (\hat{b}_i^j)', \hat{b}_i^{-j})$  where  $j \in \mathcal{N}_i \setminus \{i\}$  and  $\hat{b}_i^{-j} = \{\hat{b}_i^j\}_{j \in \mathcal{N} \setminus \{i,j\}}$ . As in Case 1, if  $[x, a]$  is a state based Nash equilibrium, then:

$$\omega_{ij} \left[ (\hat{b}_i^j)' - \omega_{ij}(\tilde{b}_i - \tilde{b}_j) \right]^2 - \omega_{ij}(\tilde{b}_i - \tilde{b}_j)^2 \geq 0 \quad (11)$$

where  $\omega_{ij} = \frac{w_i + w_j}{w_i w_j} > 0$  and  $(\tilde{v}, \tilde{b}) = f(x, a)$ .

Any state based Nash equilibrium must satisfy both (10) and (11) for any agent  $i$ ,  $(\hat{v}_i)'$ , and  $(\hat{b}_i^j)'$ . It is easy to derive that the two inequalities (10) and (11) hold for all  $(\hat{v}_i)'$ ,  $(\hat{b}_i^j)' \in \mathbb{R}$  if and only if  $\forall i \in \mathcal{N}, j \in \mathcal{N}_i$ ,

$$\beta_i = w_i \tilde{b}_i - \sum_{j \in \mathcal{N}_i} (w_i + w_j)(\tilde{v}_i - \tilde{v}_j) = 0; \quad (12)$$

$$\tilde{b}_i - \tilde{b}_j = 0. \quad (13)$$

Sum up all the  $\{\beta_i\}_{i \in \mathcal{N}}$  in the equation (12) and we have

$$\sum_{i \in \mathcal{N}} \beta_i = \sum_{i \in \mathcal{N}} w_i \tilde{b}_i - \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}_i} (w_i + w_j)(\tilde{v}_i - \tilde{v}_j) = 0.$$

Since the interaction graph is undirected and connected,  $\sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}_i} (w_i + w_j)(\tilde{v}_i - \tilde{v}_j) = 0$ . Thus  $\sum_{i \in \mathcal{N}} w_i \tilde{b}_i = 0$ . Applying equality (13), we can get  $\tilde{b}_i = 0, \forall i \in \mathcal{N}$ . Substituting  $\tilde{b} = \mathbf{0}$  into equation (12), we have

$$\sum_{j \in \mathcal{N}_i} (w_i + w_j)(\tilde{v}_i - \tilde{v}_j) = 0, \quad \forall i \in \mathcal{N} \quad (14)$$

By defining a matrix  $L = [L_{i,j}]_{n \times n}$  where

$$L_{i,j} = \begin{cases} \sum_{j \in \mathcal{N}_i \setminus \{i\}} (w_i + w_j); & \text{if } j = i \\ -(w_i + w_j); & \text{if } j \in \mathcal{N}_i \setminus \{i\} \\ 0; & \text{if } j \notin \mathcal{N}_i \end{cases} \quad (15)$$

equation (14) can be expressed as  $L \cdot \tilde{v}^T = 0$ . It is straightforward to verify that  $L \cdot \mathbf{1} = 0$ , where  $\mathbf{1} \in \mathbb{R}^n$ . Moreover, from Theorem 1 in [9], we know that  $\text{rank}(L) = n - 1$ . Thus the solution to (14) is that  $\tilde{v}^T = \zeta \mathbf{1} \in \mathbb{R}^n$ . By applying equality (6), we have  $\forall i \in \mathcal{N}, \tilde{v}_i = v^*; \tilde{b}_i = 0$ . According to Lemma 1, we know that  $[x^*, a] := [(v^*, \mathbf{0}), a]$  is also a state based Nash equilibrium, where  $v^* := (v^*, \dots, v^*)$ .

## V. GRADIENT PLAY FOR CONSENSUS PROBLEM

We will develop a distributed learning algorithm to ensure that the agents reach a state based Nash equilibrium. The proposed algorithm extends the convergence results for the algorithm gradient play [11], [22], [23] to state based games.

Consider the following algorithm: at each time  $t \geq 0$ , given the state  $x(t) = (v(t), b(t))$ , each agent  $i$  selects an action  $a_i \triangleq (\hat{v}_i, \hat{b}_i)$  according to:

$$\begin{aligned} \hat{v}_i(t) &= -\epsilon_i^v \cdot \frac{\partial J_i(x(t), a)}{\partial \hat{v}_i} \Big|_{a=0} \\ &= \epsilon_i^v \cdot 2 \left[ w_i b_i(t) - \sum_{j \in \mathcal{N}_i} (w_i + w_j)(v_i(t) - v_j(t)) \right] \\ \hat{b}_i^j(t) &= -\epsilon_{i,j}^b \cdot \frac{\partial J_i(x(t), a)}{\partial \hat{b}_i^j} \Big|_{a=0} = \epsilon_{i,j}^b \cdot 2(b_i(t) - b_j(t)) \end{aligned} \quad (16)$$

where  $\epsilon_i^v$  and  $\{\epsilon_{i,j}^b\}_{j \in \mathcal{N}_i \setminus \{i\}}$  are the stepsizes which are positive constants.

**Theorem 3.** Suppose each agent selects an action according to the gradient play algorithm in (16) at each time  $t \geq 0$ . If the stepsizes are sufficiently small, then  $[x(t), a(t)] := [(v(t), b(t)), a(t)]$  exponentially converges to  $[(v^*, \mathbf{0}), \mathbf{0}]$  where  $v^* := (v^*, \dots, v^*) \in \mathbb{R}^{mn}$ .

*Proof:* Because of space considers, we present a sketch of the proof. The general idea is to demonstrate that the proposed cost functions endow the state based game with a similar structure to the well studied potential games [24]. Analyzing this state based potential provides an upper bound of stepsizes that are needed to ensure exponential convergence.

To see this, consider the following (state based) potential function  $\phi(x, a)$

$$\phi(x, a) = \phi^v(x, a) + \phi^b(x, a) \quad (17)$$

where

$$\begin{aligned} \phi^v(x, a) &= \frac{1}{2} \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}_i} (w_i + w_j) \|v_i + \hat{v}_i - (v_j - \hat{v}_j)\|_2^2 \\ \phi^b(x, a) &= \sum_{i \in \mathcal{N}} w_i \left\| b_i - \hat{a}_i + \frac{1}{w_i} \sum_{k \in \mathcal{N}_i \setminus \{i\}} (\hat{b}_i^k - \hat{b}_i^k) \right\|_2^2 \end{aligned}$$

It is straightforward to verify that for all  $x, a, a'$ ,

$$J_i(x, a'_i, a_{-i}) - J_i(x, a_i, a_{-i}) = \phi(x, a'_i, a_{-i}) - \phi(x, a_i, a_{-i}), \quad (18)$$

Therefore,  $\phi(x, a)$  can be viewed as a potential function in a similar fashion to those in potential games [24]. Using this potential function, the gradient play rule (16) also satisfies

$$\begin{aligned} \hat{v}_i(t) &= -\epsilon_i^v \frac{\partial \phi(x, a)}{\partial \hat{v}_i} \Big|_{a=0} \\ \hat{b}_i^j(t) &= -\epsilon_{i,j}^b \frac{\partial \phi(x, a)}{\partial \hat{b}_i^j} \Big|_{a=0}. \end{aligned}$$

Using the notation  $\tilde{x} = (\tilde{v}, \tilde{b})$  where  $\tilde{v} = f^v(x, a)$  and  $\tilde{b} = f^b(x, a)$ , rewrite the potential function  $\phi(x, a)$  as

$$\begin{aligned} \phi(x, a) &= \phi(\tilde{x}, \mathbf{0}) \\ &= \frac{1}{2} \sum_{i \in \mathcal{N}} \sum_{j \in \mathcal{N}_i} (w_i + w_j) \|\tilde{v}_i - \tilde{v}_j\|_2^2 + \sum_{j \in \mathcal{N}} w_j \|\tilde{b}_j\|_2^2 \end{aligned} \quad (19)$$

The main idea of the proof is to explore the properties of the state based potential function  $\phi(x, a) = \phi(\tilde{x} := f(x, a), \mathbf{0})$ . Define  $H^x(a) := \frac{\partial^2 \phi(x, a)}{\partial^2 a}$ , which actually is a constant matrix because  $\phi(x, a)$  is a quadratic function. Thus we can simply denote it as  $H$ . Define  $C \triangleq \|H\|_2$ . By applying the same proof techniques used in [3], [11], we can show that during the gradient play learning process,  $\phi(x(t+1), \mathbf{0}) := \phi(x(t), a(t)) \leq \eta \phi(x(t), \mathbf{0})$  with  $\eta \in (0, 1)$  for each time  $t$  as long as each stepsize is smaller  $\frac{2}{C}$ . Since the minimal of  $\phi(x, \mathbf{0})$  is achieved at the weighted average point,  $x(t) := (v(t), b(t))$  converges exponentially converge to  $(v^*, \mathbf{0})$  and thus  $a(t)$  also converges exponentially to  $\mathbf{0}$ .

**Remark 1.** It is straightforward to verify that as long as each agent follows the gradient play algorithm we get the

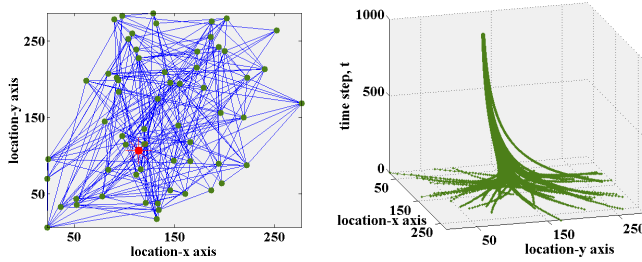


Fig. 2. The left figure shows the interaction graph: Green nodes represent agents and their positions represent their initial values  $v_i(0)$ , which are two dimensional. Blue lines represent the communication links and the red node is the weighted average point. The right figure demonstrates that the dynamics converge.  $XY$  plane is for the value information  $v$  and  $Z$  axis shows the time step. Each curve is a trajectory of one agent value  $v_i$ .

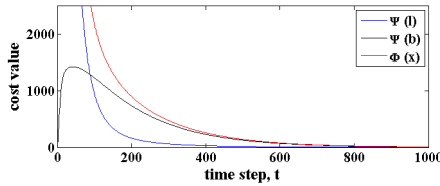


Fig. 3. Dynamics of  $\phi(x(t), a(t))$ ,  $\phi^v(x(t), a(t))$ , and  $\phi^b(x(t), a(t))$  desired convergence properties irrespective of whether the updates are synchronous or asynchronous.

## VI. SIMULATION RESULT

In this section, we simulate a weighted-average consensus problem which has 61 agents. The simulation is illustrated in Figure 2. The left figure shows the initial location information and the interaction network graph; the right figure shows the convergence of the dynamics by applying the state based game model proposed in section IV and the gradient play rules (16). Figure 3 demonstrates the evolution of  $\phi(x(t), a(t))$  (defined by equation (17)) during the gradient play learning process, which shows that  $\phi(x(t), a(t))$  converges to 0 very fast. Moreover the dynamics of the two terms  $\phi^v(x(t), a(t))$  and  $\phi^b(x(t), a(t))$  are also plotted in Figure 3. At the beginning, since the agents' values  $\{v_i\}_{i \in \mathcal{N}}$  are dispersed, the cost  $\phi(x(t), a(t))$  are mainly caused by cost  $\phi^v(x(t), a(t))$ . Gradually the agents values move closer and then the bias cost  $\phi^b(x(t), a(t))$  plays the main role.

## VII. CONCLUSION

In this paper we investigate game design for multiagent systems where the desired global behavior involved coupled constraints. First, we demonstrate that the noncooperative game framework is not suitable for dealing with such systems. Then we introduce a novel game design using the state based game framework that ensures all resulting equilibria satisfy the desired performance criterion. In essence, we introduce a state variable for equilibrium refinement. Finally we propose a distributed learning algorithm that guarantees asymptotic convergence to these desired equilibria.

While this paper primarily deals with the consensus problem, both the game design and learning design methodologies developed within this paper can be generalized to other

classes of problems. Future work involves extending the results in this paper to accommodate the needs of multiagent cooperative systems with more general coupled constraints.

## REFERENCES

- [1] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, 35(9):803–812, 1986.
- [2] E. Campos-Nàdez, A. Garcia, and C. Li. A game-theoretic approach to efficient power management in sensor networks. *Operation Research*, 56(3):552–561, 2008.
- [3] R. Olfati-saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. In *Proceedings of the IEEE*, volume 95, January 2007.
- [4] A. Tang, J. Wang, S. H. Low, and M. Chiang. Equilibrium of heterogeneous congestion control: Existence and uniqueness. *IEEE/ACM Transactions on Networking*, 15(4):824–837, October 2007.
- [5] G. Arslan, M. F. Demirkol, and S. Yüksel. A game theoretic approach to power optimization in mimo interference systems. *IEEE Transactions on Information Theory*, 2009. submitted.
- [6] V. Mhatre, K. Papagiannaki, and F. Baccelli. Interference mitigation through power control in high density 802.11. In *Proceedings of INFORCOM*, 2007.
- [7] G. Scutari, D. P. Palomar, and J. Pang. Flexible design of cognitive radio wireless systems: from game theory to variational inequality theory. *IEEE Signal Processing Magazine*, 26(5):107–123, September 2009.
- [8] D. X. Wei, C. Jin, S. H. Low, and S. Hegde. Fast tcp: motivation, architecture, algorithms, performance. *IEEE/ACM Transactions on Networking*, 14(6):1246–1259, December 2006.
- [9] R. Olfati-saber and R. M. Murray. Consensus problems in networks of agents with switching topology and time-delays. *IEEE Transactions on Automatic Control*, 49(9):1520–1533, 2004.
- [10] A. Rantzer. Dynamic dual decomposition for distributed control. In *Proceedings of American Control Conference*, St. Louis, June 2009.
- [11] L. Chen, S. H. Low, and J. C. Doyle. Random access game and medium access control design. *IEEE/ACM Transactions on Networking*, 2010.
- [12] J. R. Marden and M. Effros. The price of selfishness in network coding. In *Workshop on Network Coding, Theory, and Applications*, June 2009.
- [13] J. Marden and A. Wierman. Distributed welfare games. *Operations Research*, page Submitted, 2008.
- [14] J. R. Marden and A. Wierman. Overcoming limitations of game-theoretic distributed control. In *48th IEEE Conference on Decision and Control*, December 2009.
- [15] H. Nikaido and K. Isoda. Note on non-cooperative convex games. *Pacific J. Math*, 5:807–815, 1955. Suppl. 1.
- [16] J. B. Rosen. Existence and uniqueness of equilibrium points for concave n-person games. *Econometrica*, 33(3):520–534, 1965.
- [17] L. Pavel. An extension of duality to a game-theoretic framework. *Automatica*, 43:226–237, 2007.
- [18] S. Uryasev and R. Y. Rubinstein. On relaxation algorithms in computation of non-cooperative equilibria. *IEEE Transactions on Automatic Control*, 39:1263–1267, June 1994.
- [19] J. Krawczyk. Numerical solutions to coupled-constraint (or generalised nash) equilibrium problems. *Computational Management Science*, 4(2):183–204, April 2007.
- [20] J. R. Marden, G. Arslan, and J. S. Shamma. Cooperative control and potential games. *IEEE Transactions on Systems, Man and Cybernetics. Part B: Cybernetics*, 39, 2009.
- [21] L. S. Shapley. Stochastic games. In *Proceedings of the National Academy of Science of the United States of America*, volume 39, pages 1095–1100, 1953.
- [22] S. D. Flam. Equilibrium, evolutionary stability and gradient dynamics. *Int. Game Theory Rev*, 4(4):357–370, 2002.
- [23] J. Shamma and G. Arslan. Dynamic fictitious play, dynamic gradient play, and distributed convergence to nash equilibria. *IEEE Transactions on Automatic Control*, 50(3):312–327, March 2005.
- [24] D. Monderer and L.S. Shapley. Potential games. *Games and Economic Behavior*, 14:124–143, 1996.